



Inhaltsverzeichnis

1. Modul:Plain text/Doku	2
2. Modul:Plain text	3



Modul:Plain text/Doku

Dies ist die Dokumentationsseite für **Modul:Plain text**

Developed for producing short descriptions from text that may have markup, or other stuff that needs removing from short descriptions.

Usage

Code

```
'''[[foo|hah]]''' <span style="color:red">is</span> '''[[gah]]''''<nowiki>?</nowiki> →
```

hah is *gah*?

Using modul

```
{{#invoke:Plain text|main|1='''[[foo|hah]]''' <span style="color:red">is</span> /span> '''[[gah]]''''<nowiki>?</nowiki>}} →
```

[Vorlage:Plain text](#)

See also

[Vorlage:Navbox wikitext-handling templates](#)

Modul:Plain text

Developed for producing short descriptions from text that may have markup, or other stuff that needs removing from short descriptions.

Usage

Code

```
'''[[foo|hah]]''' <span style="color:red">is</span> '''[[gah]]''''<nowiki>?</nowiki> →
```

hah is *gah*?

Using modul

```
{{#invoke:Plain text|main|1='''[[foo|hah]]''' <span style="color:red">is</span> </span> '''[[gah]]''''<nowiki>?</nowiki>}} →
```

[Vorlage:Plain text](#)

See also

[Vorlage:Navbox wikitext-handling templates](#)

```
--converts text with wikilinks to plain text, e.g "[[foo|gah]] is [[bar]]" to "gah is bar"
--removes anything enclosed in tags that isn't nested, mediawiki strip markers (e.g. <div>
local p = {}
```

```
function p.main(frame)
    local text = frame.args[1]
    local encode = require('Module:yesno')(frame.args.encode)
    return p._main(text, encode)
end
```

```
function p._main(text, encode)
    if not text then return end
    text = mw.text.killMarkers(text)
    :gsub('&nbsp;',' ') --replace nbsp spaces with regular spaces
    :gsub('<br ?/?>',' ') --replace br with commas
    :gsub('<span.->(.-)</span>','%1') --remove spans while keeping text
    :gsub('<i.->(.-)</i>','%1') --remove italics while keeping text
    :gsub('<b.->(.-)</b>','%1') --remove bold while keeping text inside
    :gsub('<em.->(.-)</em>','%1') --remove emphasis while keeping text inside
    :gsub('<strong.->(.-)</strong>','%1') --remove strong while keeping text inside
    :gsub('<.->.-<.->','') --strip out remaining tags and the text inside
    :gsub('<.->','') --remove any other tag markup
    :gsub('%[%[%s*[Ff][Ii][Ll][Ee]%s*:.-%]%', '') --strip out files
    :gsub('%[%[%s*[Ii][Mm][Aa][Gg][Ee]%s*:.-%]%', '') --strip out user images
    :gsub('%[%[%s*[Cc][Aa][Tt][Ee][Gg][Oo][Rr][Yy]%s*:.-%]%', '') --strip out categories
    :gsub('%[%[[^]]-|','') --strip out piped link text
    :gsub('([^[^]])%[[^%]]%[[^]]-s','%1') --strip out external link
    :gsub('^%[[^%]]%[[^]]-s','') --strip out external link text
    :gsub('%[%]]%', '') --then strip out remaining [ and ]
    :gsub("''''''", "") --strip out bold italic markup
    :gsub("''''?", "") --not stripping out '''' gives correct output if
    :gsub('-+','') --remove ---- lines
    :gsub("^%s+","") --strip leading
```



```
                :gsub("%s+$", "") --and trailing spaces
                :gsub("%s+", " ") --strip redundant spaces
    if encode then
        return mw.text.encode(text)
    else
        return text
    end
end
return p
```